
Data model

Les attributs spéciaux

Les attributs spéciaux (entourés de double underscore) sont accessible depuis le type d'une classe ou l'instance d'une classe. Ceux-ci sont en en lecture seule.

Nom	Accessible sur	Objectif
<code>__name__</code>	classe	Renvoie le nom de la classe
<code>__bases__</code>	classe	Renvoie un tuple des classes parentes de la classe.
<code>__class__</code>	objet	Permet de connaître à partir de quelle classe l'objet a été créé
<code>__dict__</code>	objet	Renvoie un dictionnaire des méthodes et attributs disponibles
<code>__doc__</code>	classe & objet	Renvoie la “docstring” de la classe interrogée

Les méthodes spéciales

Les méthodes spéciales (entourés de double underscore) permettent de définir le comportement d'une classe. Celles-ci peuvent être redéfinies par la classe.

Nom	Rôle
<code>__init__(self[, ...])</code>	Permet l'initialisation des champs correct lors de la création de l'objet.
<code>__repr__(self)</code>	Envoie la représentation "officielle" en chaîne de caractères d'un objet. Cette fonction est principalement utilisée à fins de débogage
<code>__str__(self)</code>	Envoie la représentation "informelle" en chaîne de caractères d'un objet. Elle est appelée nativement par les fonctions “ <code>str(...)</code> ”, “ <code>format()</code> ” et “ <code>print()</code> ”.

<https://docs.python.org/fr/3.7/reference/datamodel.html#basic-customization>

Les méthodes spéciales

Elles permettent également de redéfinir les opérateurs de comparaison.

Nom	Opérateur
<code>__eq__(self, other)</code>	<code>==</code>
<code>__ne__(self, other)</code>	<code>!=</code>

Par défaut, il existe une relation implicite entre les opérateurs d'égalité stricte.

L'opérateur “`__ne__()`” délègue à “`__eq__()`” et renvoie le résultat inverse.

Nom	Opérateur
<code>__lt__(self, other)</code>	<code><</code>
<code>__le__(self, other)</code>	<code><=</code>

Nom	Opérateur
<code>__gt__(self, other)</code>	<code>></code>
<code>__ge__(self, other)</code>	<code>>=</code>